

**AMENDMENTS TO THE CLAIMS**

Claims 1-6, 8-13, 15-20, and 22-29 were originally pending.

Please amend claims 2-6, 9-13, and 16-19.

Please cancel claims 1, 8, 15, 20, and 22-29 without prejudice.

Accordingly, claims 2-6, 9-13, and 16-19 remain pending.

The following listing of claims replaces all prior versions, and listings of claims in the application.

1. (Canceled).

2. (Currently amended) A The method as recited in of claim 61, wherein the dynamic variable amount of time is based on a maximum amount of time that a thread can yield before needing to be scheduled for execution.

3. (Currently amended) A The method as recited in of claim 61, wherein the device is a battery powered device.

4. (Currently amended) A The method as recited in of claim 61, wherein the operating system is a Microsoft WINDOWS CE, Linux, WindRiver, QNX, or PALM operating system.

5. (Currently amended) A The method as recited in of claim 61, wherein the predetermined periodic rate is a millisecond.

1  
2       6.     (Currently amended) ~~A method as recited in claim 1: A~~  
3     computer-implemented method for providing thread scheduling in a device,  
4     the device comprising one or more hardware elements operatively coupled  
5     to an operating system comprising a plurality of program modules, the  
6     method comprising:

7         scheduling one or more threads according to a predetermined  
8     periodic rate;

9         ~~wherein the providing further comprises~~ setting a system timer to  
10     generate a notification at the predetermined periodic rate;

11         determining whether or not there are any threads to execute;

12         responsive to a determination that there are no threads to execute:

13             (a) deactivating one or more of the hardware elements and the  
14     program modules for a dynamic variable amount of time, the dynamic  
15     variable amount of time being independent of the predetermined periodic  
16     rate and being based on a sleep state of a set of threads in a sleep queue;  
17     and

18             (b) ~~wherein the deactivating further comprises~~ resetting the  
19     system timer to generate the notification after the dynamic variable amount  
20     of time has elapsed since the deactivating; and

21         wherein the method further comprises:

22             receiving the notification after the dynamic variable amount  
23     of time has elapsed since the deactivating; and

24             responsive to the receiving:  
25

1                   resetting the system timer to generate the notification  
2 at the predetermined periodic rate; and  
3                   activating the one or more of the hardware ~~modules~~  
4 elements and the program modules.

5  
6           7-8. (Canceled).

7  
8           9. (Currently amended) A The method as ~~recited in~~ of claim  
9 138, wherein the device comprises a battery powered device.

10  
11          10. (Currently amended) A The method as ~~recited in~~ of claim  
12 138, wherein the operating system comprises a Microsoft WINDOWS CE  
13 operating system.

14  
15          11. (Currently amended) A The method as ~~recited in~~ of claim  
16 138, wherein the predetermined periodic rate is a millisecond.

17  
18          12. (Currently amended) A ~~method as recited in claim 8: A~~  
19 method for providing thread scheduling in a device, the device comprising  
20 one or more hardware elements operatively coupled to an operating system  
21 comprising a plurality of program modules, the method comprising:

22               scheduling one or more threads at a predetermined periodic rate,  
23 ~~wherein the scheduling further comprises~~ comprising setting a system timer  
24 to the predetermined periodic rate, the predetermined periodic rate  
25 corresponding to a thread scheduling accuracy; and

determining whether or not there are any threads to execute;

responsive to a determination that there are no threads to execute:

(a) deactivating one or more of the hardware elements and the program modules for a dynamic variable amount of time, the dynamic variable amount of time being based on a sleep state of a set of threads in a sleep queue and independent of the predetermined periodic rate; and

(b) wherein the deactivating further comprises resetting the system timer to generate a notification after the dynamic variable amount of time has elapsed since the deactivating; and

activating the one or more of the hardware elements and the program modules only when the operating system:

(c) needs to schedule a thread for execution upon expiration of the dynamic variable amount of time since the deactivating, or

(d) upon receipt of an external event that is not a system timer event.

13. ~~(Currently amended)~~ A method as recited in claim 8: A method for providing thread scheduling in a device, the device comprising one or more hardware elements operatively coupled to an operating system comprising a plurality of program modules, the method comprising:

scheduling one or more threads at a predetermined periodic rate;

determining whether or not there are any threads to execute;

responsive to a determination that there are no threads to execute:

(a) deactivating one or more of the hardware elements and the program modules for a dynamic variable amount of time, the dynamic

1 variable amount of time being based on a sleep state of a set of threads in a  
2 sleep queue and independent of the predetermined periodic rate; and

3 (b) wherein the deactivating further comprises resetting a  
4 system timer to generate a notification after the dynamic variable amount of  
5 time has elapsed, the dynamic variable amount of time being a maximum  
6 amount of time that a thread can yield to other threads before needing to be  
7 scheduled for execution; and

8 activating the one or more of the hardware elements and the program  
9 modules only when the operating system needs to perform an action  
10 selected from a group of actions comprising scheduling a thread for  
11 execution upon expiration of the dynamic variable amount of time since the  
12 deactivating, or upon receipt of an external event that is not a system timer  
13 event, wherein the activating further comprises comprising resetting the  
14 system timer to the predetermined periodic rate to provide substantial  
15 thread scheduling accuracy.

16  
17 14-15. (Canceled).

18  
19 16. (Currently amended) A The computer-readable storage  
20 medium as recited in of claim 14, wherein the dynamic variable amount  
21 of time comprises a maximum amount of time that a thread has specified  
22 that it will yield to other threads before it needs to be scheduled for  
23 execution.  
24  
25

1           17. (Currently amended) A The computer-readable storage  
2 medium ~~as recited in~~ of claim 1915, wherein the device comprises a battery  
3 powered device.

4  
5           18. (Currently amended) A The computer-readable storage  
6 medium ~~as recited in~~ of claim 1915, wherein the operating system  
7 comprises a Microsoft WINDOWS CE operating system.

8  
9           19. (Currently amended) ~~A computer-readable storage medium as~~  
10 ~~recited in claim 15, wherein the computer-executable instructions further~~  
11 ~~comprise instructions for:~~ A computer-readable storage medium containing  
12 computer-executable instructions for scheduling threads in a device, the  
13 device including an operating system comprised of a plurality of program  
14 modules that are in turn coupled to one or more hardware elements, the  
15 computer-executable instructions comprising instructions for:

16           determining at a periodic rate whether or not there are any threads to  
17 execute; and

18           responsive to a determination that there are no threads to execute:

19           (a) deactivating one or more of the program modules and the  
20 hardware elements for a dynamic variable amount of time, the dynamic  
21 variable amount of time being independent of the periodic rate, the dynamic  
22 variable amount of time being based on a sleep state of a set of threads in a  
23 sleep queue; and

24           (b) in the deactivating, configuring a system timer to send a  
25 first timer interrupt after the dynamic variable amount of time has elapsed,

1 the dynamic variable amount of time being a maximum amount of time that  
2 a first thread can yield to a second thread before the first thread needs to be  
3 executed;

4 responsive to receiving the first timer interrupt:

5 (ac) configuring the system timer to send a second timer  
6 interrupt at the periodic rate; and

7 (bd) activating the one or more of the program modules and  
8 the hardware elements to determine if there are any threads to execute.

9  
10 20-29. (Canceled).